Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2025 SCPE. Volume 26, Issues 5, pp. 2002–2015, DOI 10.12694/scpe.v26i5.4818

ENHANCED PRE-PROCESSING STRATEGIES FOR ACCURATE DIABETES PREDICTION IN HEALTHCARE USING NOVAL METHOD: ANN+LDA

SOUMYA K N* AND RAJA PRAVEEN N†

Abstract. In the recent past, so many chronic diseases have been emerging and spreading in the world and even in the developing and advanced countries as well. One of such serious chronic diseases is Diabetes Mellitus that covers and impacts the health of people from early age. Nevertheless, the available Machine Learning (ML) and Deep Learning (DL) approaches are unable to provide good predictions in patients relating to diabetes. In addition, this study evaluated the proposed pre-processing procedure on large datasets for diabetes prediction that contained outliner detection and removal, missing values imputation, and standardization, to improve diabetes ascertainment. This research evaluated the proposed pre-processing procedure on a large set of data by outlier identification and removal, missing values imputation and data standardization were done to improve diabetes forecast. To ensure rapid and accurate classification of diabetes, the researchers employed and initialized an Artificial Neural Network (ANN). Data was gathered from the PIMA Dataset and North California State University (NCSU). Following this, Bivariate filter was applied to sort out features which were relevant. The selected features were subsequently subjected to Pearson correlation towards feature set refinement considering a threshold below which features were eliminated and only the most effective features selected. From the results it was evident that the proposed approach was significantly better than the existing methods in terms of accuracy as it achieved a classification accuracy of around 93% as opposed to the other methods

Key words: Artificial Neural Network, Bivariate filter, Diabetes mellitus, Pearson correlation, Standardization.

1. Introduction. Diabetes is a chronic condition that may soon threaten the health of the entire population. The International Diabetes Federation reported that currently there are 382 million diabetics across the world. Unfortunately, the forecast is for this number to rise to 592 million by the year 2035, and essentially doubling. It is a disease in which blood glucose levels are high, making it a concern for much of the population as well as healthcare systems [1]. Early prediction of diseases like diabetes can play a crucial role in controlling and potentially saving human lives. This research aims to predict diabetes by analyzing various disease-related attributes. The study utilizes the Pima Indian Diabetes Dataset and applies various machine learning (ML) classification and ensemble techniques. These methods aim to accurately predict diabetes occurrence, facilitating early intervention and improving healthcare outcomes [2]. Machine Learning (ML) is an explicit training method used to efficiently gather knowledge by building various classification and ensemble models from large datasets. This approach can be employed to develop predictive models for diabetes using extensive data repositories [3]. These learning models generate insights from large datasets, which can then be applied to new data for prediction and analysis. In recent years, ML has gained significant attention in healthcare, particularly for tasks such as disease diagnosis, including diabetes prediction.

In the context of disease diagnosis, ML enables the development of systems that can effectively assist physicians in identifying diseases [4]. Rapid advancements in Artificial Intelligence (AI), especially in ML and computer vision, have led to applications that automate complex, intelligence-demanding tasks in healthcare. These technological developments are particularly valuable for tasks such as diabetes prediction and diagnosis, where analysis of large datasets and recognition of subtle patterns are crucial.

This research focused on training an ML model to predict the progression from pre-diabetes to diabetes. The model utilized Electronic Medical Records (EMR), incorporating both historical and current patient data. The researchers thoroughly described the model's development and validation process, using data from The

*School of Computer Science and Engineering, JAIN (Deemed to be University) Bangalore, Karnataka, India (Soumya.kn16@gmail.com)
†Department of Computer Science and Engineering, Faculty of Engineering and Technology, JAIN (Deemed-to-be University), Bengaluru, 562112, India (p.raja@jainuniversity.ac.in)

Health Improvement Network (THIN) database for both internal and external validation [5]. This approach aligns with the broader ML process for diabetes prediction, which involves gathering pertinent data from EMR systems like THIN, including comprehensive medical records and lifestyle factors. From the collected EMR data, researchers extract relevant features and train an ML algorithm to build a predictive model for diabetes progression. The resulting model then analyzes new patient data to evaluate the risk of diabetes development based on specific input variables [6]. This approach equips healthcare systems with a powerful tool for early diabetes risk assessment, seamlessly integrating software engineering technology with ML techniques to support proactive patient care. By providing timely risk evaluations, the system enables healthcare providers to implement targeted interventions and personalized management strategies, potentially improving patient outcomes and reducing the overall burden of diabetes on healthcare systems.

The diabetes management system aims to determine optimal nutrition requirements and provide tailored meal recommendations for patients. The system also sends timely medication reminders, enhancing overall healthcare management and support [7]. Beyond these specific features, ML plays a crucial role in broader diabetes care. It assists healthcare professionals in early detection and intervention for high-risk individuals, facilitating the development of personalized treatment plans and lifestyle adjustments. Furthermore, ML is instrumental in developing decision support systems for diabetes management, providing real-time insights and recommendations to healthcare providers. This comprehensive approach, combining personalized patient support with advanced clinical decision-making tools, has the potential to significantly improve diabetes outcomes and quality of life for patients.

Additionally, these predictive models are essential for population health studies, as they enable the identification of risk factors and the implementation of preventive measures on a broader scale [8]. Benefits of using ML for diabetes prediction include early detection, personalized risk assessment, optimized treatment planning, reduced healthcare costs, improved patient management, enhanced quality of life, and the potential to prevent complications. ML algorithms can process vast amounts of data, identify patterns, and deliver accurate predictions, enabling timely interventions and better outcomes for those at risk of developing diabetes [9]. However, drawbacks include the possibility of false positives or negatives and dependence on accurate data input.The models have limited interpretability and pose a risk of over-reliance on technology without sufficient clinical judgment. Overcoming these challenges demands careful planning, stakeholder involvement, and a well-executed transition strategy to effectively leverage the benefits of ML while minimizing disruptions to clinical workflows [10-13]. The suggested method changed the feature scale led to varying coefficients, rendering the magnitude coefficient an unsuitable choice for determining feature importance in the model.

Isfafuzzaman Tasin et al. [14] implemented a six ML approaches, including decision tree, SVM, Random Forest, Logistic Regression, KNN, and various ensemble techniques, to achieve the most accurate diabetes prediction. It also utilized a semi-supervised model with extreme gradient boosting to predict insulin features from a private dataset. To address the issue of class imbalance, SMOTE and ADASYN techniques was implemented. Moreover, it developed a user-friendly Android smartphone application and a suggested framework that allows users to input diverse features for instantaneous diabetes prediction. However, the suggested method, need to further enhance by incorporating fuzzy logic techniques and optimization approaches into the ML models.

Annamalai R and Nedunchelian [15] have developed an Optimal Weighted based Deep Artificial Neural Network (OWDANN) algorithm, for predicting diabetes mellitus disease and estimating its severity level. The system comprises two distinct phases: disease prediction and severity level estimation. In the disease prediction phase, the Pima dataset undergoes preprocessing to enhance data quality. Subsequently, relevant features were extracted from the preprocessed data, and the classification step employs the OWDANN algorithm. This approach effectively addresses noise and efficiently restores corrupted data, leading to improved prediction accuracy. However, OWDANN requires further modifications and need to enhance a wide range of scenarios and provide more comprehensive predictions for diabetes-related complications.

The proposed method aims to address the limitations found in existing diabetes prediction approaches. By enhancing data quality and eliminating inconsistencies, the suggested approach aims to significantly improve the accuracy and reliability of prediction models. As a result, the diabetes predictions generated are expected to be more robust and effective.

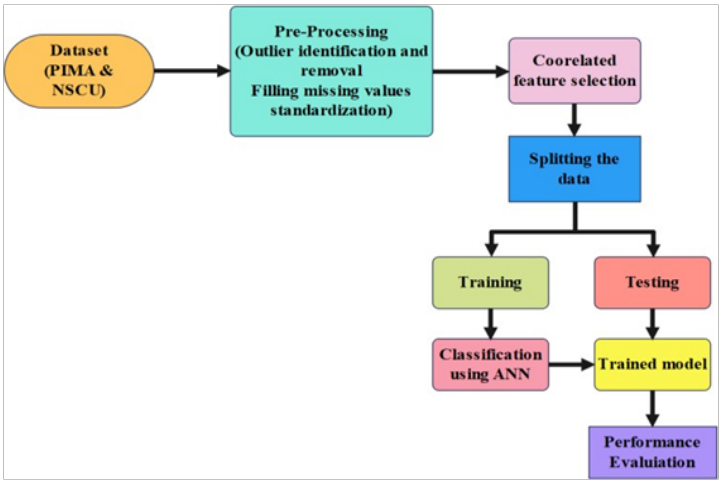

Fig. 2.1: Flow diagram of the proposed method

2. Methodology. The purpose of the ITSOA has to improve the accuracy of EEG based emotion classification. The process involved in various stages of classifying diabetes are data acquisition, pre- processing, feature selection and classification of diabetes.

The initial step involves acquiring data from a publicly available dataset, followed by a pre- processing phase to eliminate irrelevant or inappropriate features. Subsequently, a feature selection process is applied to choose relevant and non-redundant features.

Finally, an efficient classification is conducted using an ANN classifier to achieve accurate predictions. These steps aim to enhance the accuracy and reliability of the prediction models by improving data quality and removing inconsistencies, leading to more robust and effective diabetes predictions. The block diagram of the suggested method is illustrated in Fig. 2.1.

2.1. Data collection. In this research, the raw data is obtained from two publicly available datasets such as PIMA Dataset [16] and North California State University (NCSU) [17] dataset. The description of the mentioned dataset is mentioned as follows:

PIMA: The Pima Indian Diabetes dataset has been a standard benchmark for diabetes classification research due to its binary outcome variable, making it suitable for supervised learning, especially logistic regression. However, researchers have explored various ML algorithms to build classification models using this dataset, allowing for diversity and avoiding reliance on a single type of model.

NCSU: The diabetes prediction dataset used in this study was obtained from NC State University and consists of 442 instances with 10 attributes. The feature set comprises age and sex, and for the analysis, one feature set was selected.

2.2. Data Pre-processing. After the stage of data collection, in this research, the preprocessing step of the proposed framework aims to transform the data into a processed format without complexities. It involves outlier identification and removal to eliminate extreme data points, filling missing values to ensure complete datasets, and standardization to normalize variables. The process of the proposed method is briefly outlined as follows;

Outlier identification and removal. The purpose of outlier identification and removal using pre-processing for diabetes prediction is to improve model accuracy by eliminating extreme data points by improving the model's performance, ensuring more reliable predictions for better outcomes. It is a crucial step in data preparation to enhance the accuracy of predictive models. Outliers are data points that significantly deviate from the majority, potentially distorting the model's performance. Through the application of pre-processing techniques, outliers can be detected and effectively eliminated from the dataset. In the context of diabetes prediction, this

procedure enhances the model's robustness and generalization capabilities by reducing erroneous data. The conventional approach for identifying and removing outliers in multivariate data analysis involves measuring the distance of each observation using Mahalanobis distance, as depicted in Equ.2.1.

$$D_M = (X - \bar{X})^T S^{-1} (X - \bar{X}) \tag{2.1}$$

where D_M=Scaler matrix, X=Vector, $\bar{X}$=Mean of matfrix X, S^{-1}=Inverse of matrix. The observations associated with large values of DM are classified as outliers and then discarded. The Mahalanobis distance can be related to the principal components: it can be shown, in fact, that the sum of squares of the PC, standardized by the eigenvalue size, equals the Mahalanobis distance for observation I was illustrated in Equ.2.2.

$$\sum_{k=1}^{Q} \frac{Z_i k^2}{l_k} = \frac{Z_i 1^2}{l_1} + \frac{Z_i 2^2}{l_2} + ... + \frac{Z_i Q^2}{l_Q} = D_{M,i} \tag{2.2}$$

where, Q =upper limit, Z_{i1}^2=Specific instance of Z, $D_{M,i}$=Results of summerization, Z_{ik}^2=weights.

In high-dimensional datasets, some outliers may not be apparent when examining individual dimensions, making them undetectable through univariate analysis. Consequently, a multivariate approach is necessary. In this regard, Principal Component Analysis (PCA) is an excellent tool for effectively identifying and removing outlier observations.

Filling missing values. The goal of filling in missing values during pre-processing for diabetes prediction is to create a complete dataset that is ready for analysis, as missing data can adversely affect the performance of predictive models. Imputing missing values enables the model to make more accurate predictions, leading to better diabetes diagnosis and treatment. Missing data can occur for various reasons, including data entry errors or incomplete information. To ensure the integrity and accuracy of the predictive model, these gaps are filled with estimated or imputed values using different techniques.Common techniques for handling missing values include mean, median, or mode imputation, which use the central tendency of the available data to replace missing entries. More advanced methods, such as regression imputation or K-Nearest Neighbors (KNN) imputation, leverage the relationships between variables in neighboring data points. In the proposed framework, missing or null values were filled using the mean values of the attributes instead of being discarded, as shown in Equ.2.3. Mean imputation is advantageous because it fills continuous data without introducing outliers.

$$(x) = f(x) = \{(mean(x), if x = null/missedx, otherwise \tag{2.3}$$

where x is the instances of the feature vector that lies in n-dimensional space, $x \epsilon R$.

Standardization. Standardization is a key preprocessing step to ensure that features are on a comparable scale, which can improve the performance of machine learning models. We standardize these datasets:

Steps for Standardization:

1. Calculate the Mean and Standard Deviation: For each feature in the dataset, compute the mean and standard deviation.

$$Mean(\mu) = \frac{1}{N} \sum_{i=0}^{N} xi \tag{2.4}$$

$$Standard\ deviation = \sqrt{\frac{1}{N} \sum_{i=1}^{N} [x_i - \mu]^2} \tag{2.5}$$

2. Transform the Features: Subtract the mean from each feature value and divide by the standard deviation:

$$Standardized\ Value = \frac{x_i - \mu}{\sigma} \tag{2.6}$$

The Pima Indians Diabetes dataset often includes features like glucose levels, blood pressure, skin thickness, insulin levels, and BMI. Here's a brief outline of standardization for this dataset:

1. Compute the mean and standard deviation for each feature
2. Apply the transformation to each feature to ensure they have a mean of 0 and a standard deviation of 1.

2.3. Feature selection. The first pre-processing step yields the pre-processed output, which is then used as input for the feature selection stage that follows. By choosing the most pertinent features that make the diabetes classification process easier, feature selection seeks to increase classification accuracy. The present study employs the Bivariate statistics technique for feature selection, which effectively picks pertinent and suitable characteristics from extensive datasets such as PIMA and NCSU [18-24]. The Bivariate filter is employed for feature extraction, effectively integrating heterogeneous data layers to address uncertainties in the input data. Additionally, this filter utilizes a certainty factor to identify relevant features, and its evaluation is carried out using the next equation:

$$CF = \frac{PP_a - PP_s}{PP_a(1 - PP_s)}, if PP_a \geq PP_s else \frac{PP_a - PP_s}{PP_s(1 - PP_a)}, if |PP_a < PP_s| \tag{2.7}$$

where the conditional probability of CF is denoted as PP_a and the prior probability of the selected features are represented as PP_s. The value of PP_a and PP_s is evaluated using the Equ.2.8 and Equ.2.9 respectively.

$$PP_a = PS|B \tag{2.8}$$

$$PP_s = PS \tag{2.9}$$

where the conditional probability unit of **B** is represented as $PS|B$. Positive results indicate an increase in the certainty value, whereas negative results signify a decrease in the certainty value. Furthermore, the features are extracted using the Weight of Evidence (WoE) based on the Bayesian probability approach, utilizing weights to determine their significance. The positive and negative weights of the features are evaluated using two parameters, $W+$ and $W-$, as represented in Equ.2.10 and Equ.2.11.

$$W^+ = ln\frac{PB|A}{PB|\bar{A}} \tag{2.10}$$

$$W^- = ln\frac{P\bar{B}|A}{P\bar{B}|\bar{A}} \tag{2.11}$$

The logarithm and probability values are represented as P and ln respectively. The features selected using the bivariate filter method are then used as input for Pearson correlation, which identifies effective features based on a specified threshold value.

Pearson Correlation. To enhance the relationship between Pearson correlation and diabetic characteristics, the parameters are optimized to remove redundant information. The Pearson correlation coefficient is used to assess linear relationships between random variables. Equ.2.12 illustrates the linear correlation between two continuous variables

$$r_{xy} = \frac{\sum(x_i - \bar{x})\sum(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2}\sqrt{\sum(y_i - \bar{y}2)}} \tag{2.12}$$

If rxy= 1, x and y are a totally positive correlation, If rxy = 0, the linear correlation between x and y is not obvious and when rxy = -1, x and y are a totally negative correlation.

The Pearson correlation, with an R-value of 0.12, exhibits a less significant effect on Diabetes. The relationship between certain features and diabetes is found to be moderate (r = 0.33, r = - 0.42, r = 0.23). It is important to note that correlation does not imply causation. Utilizing this information, relevant features strongly associated with diabetes are identified and selected as input variables for precise disease classification. The output of Pearson correlation is then fed into the classification process to identify cases of type II diabetes.

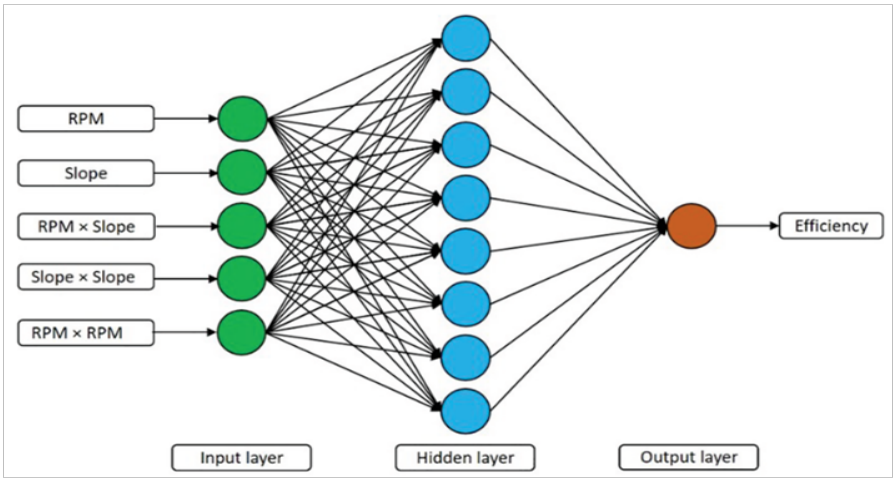

Fig. 2.2: Architecture of ANN model

2.4. Classification using ANN. Following the feature selection phase, the PIMA and NCSU datasets were used for the classification. Artificial Neural Networks (ANN), one of the machine learning (ML) algorithms used for classification, are renowned for producing results that are more accurate than those obtained using other methods. An ANN is made up of one or more hidden layers where information is processed by neurons. In order to get better outcomes, every node serves as an activation node that classes the output of artificial neurons. Numerous factors, such as the choice of hidden units for each layer, can be tuned in order to reduce the training error. These units determine the name and size of the layers, enabling users to customize the neural network's structure. The ANN algorithm is adept at finding minima, controlling variance, and subsequently updating the model's parameters, as expressed in next equation.

$$\theta = \theta - \eta * De; taJ(\theta) \tag{2.13}$$

The learning rate is another critical parameter in ANN, responsible for adjusting the weights at each step and playing a crucial role in the model's learning process. It must be carefully chosen, as a learning rate that is too high may hinder the selection of minima, while one that is too low can slow down the learning speed. Commonly selected learning rate values are in the power of 10, such as 0.001, 0.01, 0.1, and 1. In this model, the learning rate is set to 0.1. Figure 2.2 represents the architecture of ANN model.

The ANN-LDA algorithm in shown in Algorithm 1.

Start with new weights each time to perform training. Next, the LDA is used to minimize the dimensionality of the input characteristics, simplify the operations of the neural network, and optimize the classes. The weights are continuously adjusted by the model based on the error ascertained by the error Calculation. Following that, the ANN learns to decrease error by modifying the rule weights. The process is repeated until the error no longer decreases dramatically; at that point, the model is considered to have learned the relationships discovered in the data. After training each LDA-transformed feature via the ANN, apply the activation function to the weighted sum to provide an output for each neuron.

Determine the efficient result of the output layer, which gives the predicted risk of diabetes. Find the gradient of the loss function with respect to each network weight. Include the LDA-generated probabilities in the weight update process and adjust the weights to emphasize the components that LDA identified.

The network converges when weight changes in certain threshold when the maximum number of iterations is reached. Update the weights through the whole training dataset many times. Make use of both forward and backward propagation methods to adjust weights based on the difference between the actual and labels. This method effectively combines the dimensionality reduction and class separation strengths of LDA with the nonlinear modelling capabilities of an ANN.

Algorithm 1 ANN-LDA algorithm

1. **Step 1:** Data Preprocessing
   Input: Dataset (X, y)
   Output: Processed data ($X_processed$)
   (a) Normalize the dataset X (mean = 0, variance = 1)
   (b) Handle missing values (imputation or removal)
   (c) Convert categorical variables to numerical (one-hot encoding)
2. **Step 2:** Dimensionality Reduction using LDA
   Input: Processed data ($X_processed, y$)
   Output: Reduced dimension data (X_LDA)
   (a) Apply LDA to $X_processed$ with target labels y
   (b) Compute the linear discriminants for class separation
   (c) Project $X_processed$ onto the LDA components to get X_LDA
3. **Step 3:** Initialize the ANN Model
   Input: X_LDA, hyperparameters ($learning_rate$, epochs, $hidden_layers$, etc.)
   Output: Initialized ANN model (weights, biases)
   (a) Define the ANN structure with input layer size = X_LDA dimensions
   (b) Initialize weights and biases for all layers
   (c) Choose activation function (e.g., ReLU, Sigmoid)
   (d) Define the loss function (e.g., Cross-Entropy Loss)
4. **Step 4:** Train the ANN Model
   Input:X_LDA, y, ANN model
   Output: Trained ANN model (optimized weights and biases)
   (a) For each epoch in range(epochs):
   For each batch in the training data:
   Forward Pass:
   - Compute the output of each layer (activations)
   - Calculate the predicted output using the final layer
   Compute Loss:
   - Compare the predicted output with the true labels (y)
   -Calculate the loss using the chosen loss function
   Backward Pass (Backpropagation):
   - Calculate the gradient of the loss with respect to weights and biases
   - Update the weights and biases using the learning rate
   End For
5. **Step 5:** Model Evaluation
   Input: Test data (X_test, y_test), Trained ANN model
   Output: Performance metrics (accuracy, precision, recall, F-score)
   (a) Apply LDA transformation to X_test
   (b) Forward pass X_test_LDA through the trained ANN model
   (c) Calculate the predicted output for test data
   (d) Compare predictions with true labels y_test
   (e) Compute performance metrics (accuracy, precision, recall, F-score)
6. **Step 6:** Output the Model and Performance Metrics
   Return: Trained ANN-LDA model, Performance metrics

Table 3.1: Performance comparison of classifiers for PIMA dataset

Classifiers	Accuracy (%)	F-measure (%)	Recall (%)	Precision (%)
LR	78	75	77	74
SVM	87	78	85	73
KNN	76	72	73	72
ANN	82	81	86	77
DT	76	74	74	74
ANN-LDA	93	84	88	81

3. Result Analysis. In this section, the results obtained from the proposed is evaluated to obtain the results based on diabetes classification. The result section is sub-sectioned to performance analysis and the comparative analysis. The performance analysis involves assessing the efficiency of the proposed approach on two distinct datasets, namely PIMA and NCSU. For the comparative analysis, the proposed approach's effectiveness is evaluated against existing approaches documented in the literature. The evaluation metrics encompass accuracy, precision, recall, and f-measure, which are computed using the equations (12-15) as provided below.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.3}$$

$$F1measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3.4}$$

where the TP,FP,TN,FN is the True Positive, False Positive, True Negative and False Negative respectively.

3.1. Experimental setup. The proposed ANN-LDA classification approach was implemented on a system with the following specifications: Anaconda Navigator 3.5.2.0 (64-bit), Python 3.7 software, Windows 10 (64-bit) operating system, Intel Core i7 processor, and 16 GB of random-access memory.

3.2. Performance analysis of PIMA Dataset. The performance analysis of the PIMA dataset includes evaluating ML models for diabetes prediction, exploring pre-processing, Feature selection and Classification models for effectiveness, thus enabling valuable insights for healthcare applications and improving diabetes diagnosis and management [25-27]. Figures 3.1 to 3.4 presents the different performance measures. Table 3.1 shows Performance comparison of classifiers for PIMA dataset.

3.3. Performance analysis of PIMA Dataset for Pre-processing. In this subsection, we evaluate the performance of the proposed approach using various classifiers, including Naïve Bayes, KNN, Support Vector Machine (SVM), and Random Forest. The evaluation is conducted on the PIMA dataset, and the results are presented in Table 3.1 and Table 3.2.

Table 3.1 displays the results obtained from the proposed method on the PIMA dataset without applying any pre-processing techniques, while Table 3.2 shows the results after applying pre- processing techniques. Additionally, Figure 3.5 provides a graphical representation of the performance analysis for the PIMA dataset. Table3.3 presents Performance analysis for after pre-processing techniques.

The results from Table 3.1 and Table 3.3 demonstrate that the proposed method serves as an excellent classifier for distinguishing diabetic patients in the PIMA dataset. The performance of the proposed classification approach outperforms existing methods in terms of overall metrics, particularly in accuracy.

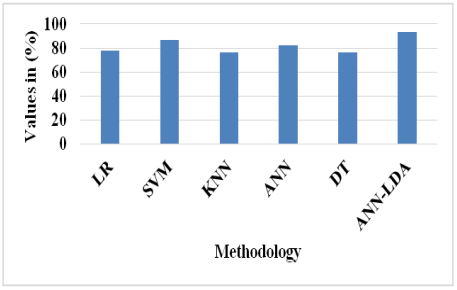

Fig. 3.1: Graphical representation of classification performance of accuracy for PIMA dataset

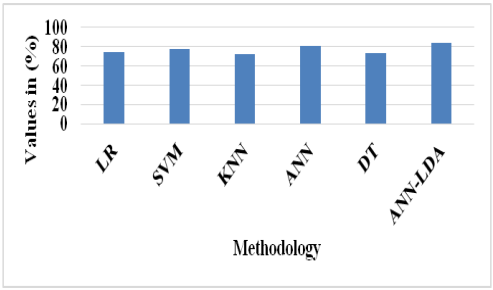

Fig. 3.2: Graphical representation of classification performance of F1-measure for PIMA dataset

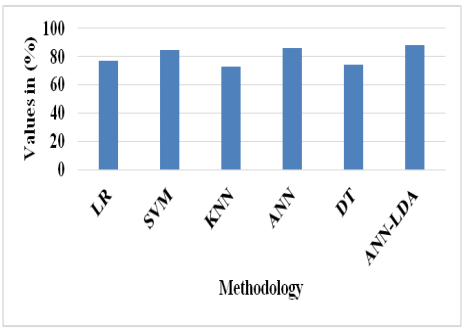

Fig. 3.3: Graphical representation of classification performance of Recall for PIMA dataset

Table 3.2: PIMA dataset for without pre-processing techniques

Methods	Accuracy (%)		
	Mean	Median	Most Frequent
Naïve Bayes	74.21	68.03	74.52
SVM	75.30	74.25	76.35
Random Forest	76.32	74.54	73.21
Proposed	70.16	58.23	61.98

3.4. Performance analysis of PIMA Dataset for feature selection. Table 3.4 presents the results obtained from the proposed method applied to the PIMA dataset using various feature selection techniques. The

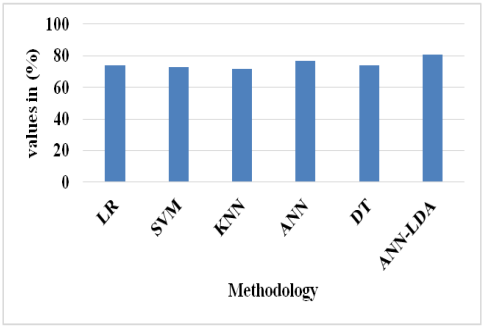

Fig. 3.4: Graphical representation of classification performance of precision for PIMA dataset

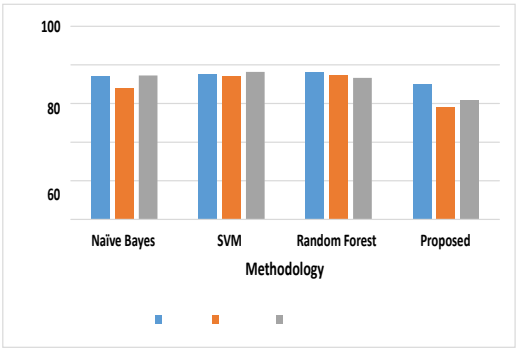

Fig. 3.5: Graphical representation of PIMA dataset for without pre-processing techniques

Table 3.3: Performance analysis for after pre-processing techniques

Missing value strategy	Z- Score	Minmax Scalar
Mean	74.65	83.45
Median	60.19	81.10
Most Frequent	64.15	80.26

Table 3.4: Performance analysis of Feature selection for PIMA Dataset

Classifier	Accuracy for Testing (%)	Accuracy for Validation (%)
SVM	75.37	81.41
Random Forest	77.23	82.89
Correlated function	78.25	85.21

dataset is divided into training and test sets in a ratio of 70% and 30%, respectively. The training set consists of 70% of the data randomly chosen, while the remaining 30% is allocated to the testing set. This specific split ratio was determined after exploring various combinations and has proven to be efficient in achieving better results.

According to the data presented in Table 3.4, the correlated function outperforms the SVM and Random Forest classifiers in terms of training and testing accuracy, following data pre- processing. Furthermore, both classifiers achieve similar validation accuracy. Notably, the correlated function demonstrates a substantially

Table 3.5: Comparing the performance of the classifiers for PIMA dataset

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
KNN	75.11	71.61	72.32	72.64
LR	76.25	73.84	76.74	74.45
DT	75.25	73.16	73.45	73.73
SVM	86.59	72.18	84.27	83.38
ANN	90.66	80.32	87.73	85.35

Table 3.6: NCSU dataset for without pre-processing techniques

Methods	Accuracy (%)		
	Mean	Median	Most Frequent
Naïve Bayes	73.31	69.06	73.11
SVM	74.36	73.35	75.22
Random Forest	76.66	73.36	72.35
Proposed	70.13	57.22	60.46

Table 3.7: Performance analysis for NCSU dataset after pre-processing techniques

Missing value strategy	Z- Score	Minmax Scalar
Mean	73.21	82.47
Median	60.32	80.65
Most Frequent	63.51	80.51

higher true negative rate, implying a more accurate prediction capability.

3.4.1. Performance analysis of PIMA Dataset for Classification. The performance evaluation of mentioned classifiers was conducted using the PIMA dataset, as shown in Table 3.5. Additionally, Table 3.5 displays the results obtained from the proposed approach for the same PIMA dataset.

Table 3.5 demonstrates that the proposed ANN serves as a highly effective classifier for distinguishing diabetic patients within the PIMA dataset. The proposed classification approach outperforms existing methods across various metrics. Additionally, the classification accuracy of the proposed ANN reaches an impressive 90.66%, surpassing the accuracies of other classifiers, such as KNN of 75.11%, LR of 76.25%, DT of 75.25%, and SVM of 86.59% respectively.

3.5. Performance Analysis of NCSU dataset. The performance analysis of the NCSU dataset involves assessing various ML models for predicting diabetes. It also encompasses exploring the effectiveness of pre-processing, feature selection, and classification models. These findings provide valuable insights for healthcare applications, leading to enhancements in diabetes diagnosis and management.

3.5.1. Performance analysis of NCSU Dataset for Pre-processing. Within this sub-section, the proposed approach's performance is assessed using various classifiers, including Naïve Bayes, KNN, SVM, and Random Forest. The evaluation is based on the NCSU datasets, and the results are presented in Table 3.6 and Table 3.7. These tables illustrate the outcomes of the proposed method on the NCSU dataset, both without and after employing pre-processing techniques. Furthermore, a graphical representation of the performance analysis for the NCSU dataset is provided in Figure 3.6.

The results from Table 3.6 and Table 3.7 demonstrate that the proposed method serves as an outstanding classifier in accurately identifying diabetic patients within the NCSU dataset. When compared with existing classification methods, the proposed approach achieves superior results in overall metrics, particularly in terms of accuracy.

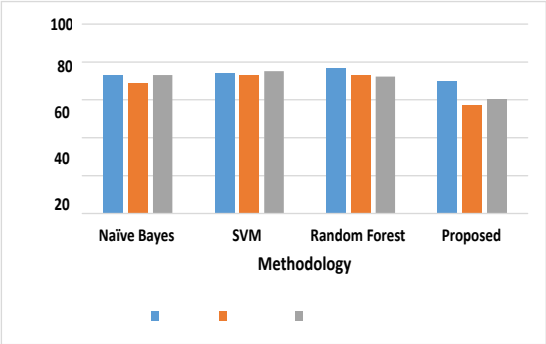

Fig. 3.6: Graphical representation of the NCSU for without pre-processing techniques

Table 3.8: Performance analysis of Feature selection for NCSU Dataset

Classifier	Accuracy for Testing (%)	Accuracy for Validation (%)
SVM	74.32	80.25
Random Forest	76.45	81.94
Correlated function	78.86	85.37

Table 3.9: Comparing the performance of the classifiers for NCSU dataset

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
KNN	74.36	72.94	71.33	71.62
LR	75.65	74.34	75.42	73.31
DT	74.12	74.16	73.97	72.30
SVM	85.10	71.88	83.55	82.11
ANN	90.37	80.34	88.52	86.19

3.5.2. Performance analysis of NCSU Dataset for feature selection. The Table 3.8 presents the results obtained from the proposed method applied to the NCSU dataset using various feature selection techniques. The dataset is divided into training and test sets at a ratio of 70% and 30%, respectively. This split is determined after exploring various combinations, proving its efficiency in achieving optimal performance.

Table 3.8 reveals that after data pre-processing, the training accuracy and testing accuracy of the correlated function surpass those of the SVM and Random Forest classifiers. Additionally, both classifiers achieve similar validation accuracy. These results indicate that the correlated function exhibits a significantly higher true negative rate, highlighting its superior correctness in predictions.

3.5.3. Performance analysis of NCSU Dataset for Classification. The performance evaluation of the recommend classifiers was conducted using the NCSU datasets, as depicted in Table 3.9. Additionally, Table 3.9 presents the results obtained from the proposed approach for the NCSU dataset.

Table 3.9 demonstrates that the proposed ANN serves as an outstanding classifier for accurately classifying diabetic patients within the NCSU dataset. The proposed classification approach achieves superior results in overall metrics compared to existing classification methods. Notably, the classification accuracy of the proposed ANN reaches 90.37%, which is significantly higher than the accuracies of existing classifiers, such as KNN of 74.36%, LR of 75.65%, DT of 74.12%, and SVM of 85.10%.

3.6. Comparative analysis. Comparative analysis refers to the Comparison of data to identify similarities and differences for meaningful insights or decision-making. In this subsection, the classification approach's performance is assessed by comparing it with existing approaches listed in related works. Evaluation is based

Table 3.10: Comparative analysis of various classifier for PIMA dataset

Classifiers	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
2GDNN [12]	97.93	98.11	97.23	97.95
Six- ML [14]	88.05	82.02	80.56	81.21
OWDANN [15]	98.97	97.02	93.84	94.04
Proposed	98.99	98.15	97.25	97.96

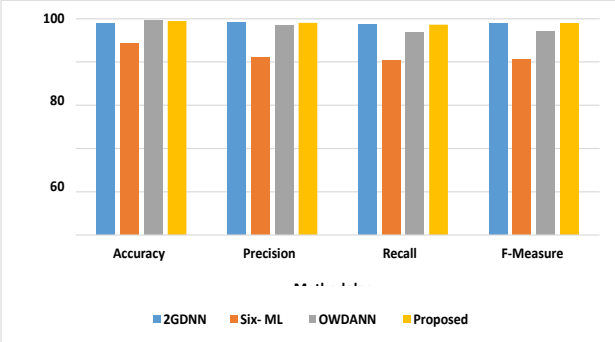

Fig. 3.7: Graphical representation of comparative analysis for PIMA dataset

on performance metrics such as accuracy, precision, recall, and F-measure score. The results obtained from evaluating the proposed approach for the PIMA dataset are presented in Table 3.10.

The graphical representation of the comparative analysis for PIMA dataset was illustrated in figure 3.7.

Table 3.9 and Figure 3.2 demonstrate that the proposed classification approach outperformed other methods in overall performance metrics. The accuracy achieved by the proposed approach is 98.99%, significantly higher than the Twice Growth Deep Neural Network (2GDNN) (97.93%), Six ML methods (88.05%), and Optimal Weighted based Deep Artificial Neural Network (OWDANN) (98.97%).

4. Conclusion. The research introduces a pre-processing approach involving outlier identification, missing value filling, and standardization to enhance diabetes Mellitus prediction accuracy. The proposed method utilizes an ANN with optimized weight initialization for effective diabetes classification. The approach's performance is evaluated on both PIMA and NCSU datasets using accuracy, precision, recall, and F-measure metrics. Following the Bivariate filter-based feature selection stage, relevant features are selected, and the chosen features undergo Pearson correlation analysis using a threshold value. The resulting effective features are then utilized as input for the ANN classifier, performing the final classification. The proposed approach outperforms existing methods in overall metrics, with an accuracy of 98.99%, surpassing 2GDNN, Six ML methods, and OWDANN of 7.93%, 88.05%, and 98.97% respectively. Future work can explore incorporating meta-heuristic algorithms to further enhance accuracy by selecting appropriate features.

REFERENCES

[1] Rani, K.J., 2020. Diabetes prediction using machine learning. International Journal of Scientific Research in Computer Science Engineering and Information Technology, 6, pp.294-305.
[2] Soni, M. and Varma, S., 2020. Diabetes prediction using machine learning techniques. International Journal of Engineering Research & Technology (Ijert) Volume, 9.
[3] Kaul, S. and Kumar, Y., 2020. Artificial intelligence-based learning techniques for diabetes prediction: challenges and systematic review. SN Computer Science, 1(6), p.322.
[4] Assegie, T.A. and Nair, P.S., 2020. The performance of different machine learning models on diabetes prediction. International journal of scientific & technology research, 9(01).
[5] Cahn, A., Shoshan, A., Sagiv, T., Yesharim, R., Goshen, R., Shalev, V. and Raz, I., 2020. Prediction of progression from

pre-diabetes to diabetes: Development and validation of a machine learning model. Diabetes/metabolism research and reviews, 36(2), p.e3252.

[6] Sowah, R.A., Bampoe-Addo, A.A., Armoo, S.K., Saalia, F.K., Gatsi, F. and Sarkodie-Mensah, B., 2020. Design and development of diabetes management system using machine learning. International journal of telemedicine and applications, 2020.

[7] Vehí, J., Contreras, I., Oviedo, S., Biagi, L. and Bertachi, A., 2020. Prediction and prevention of hypoglycaemic events in type-1 diabetic patients using machine learning. Health informatics journal, 26(1), pp.703-718.

[8] Nibareke, T. and Laassiri, J., 2020. Using Big Data-machine learning models for diabetes prediction and flight delays analytics. Journal of Big Data, 7, pp.1-18.

[9] Jaiswal, V., Negi, A. and Pal, T., 2021. A review on current advances in machine learning based diabetes prediction. Primary Care Diabetes, 15(3), pp.435-443.

[10] Ramesh, J., Aburukba, R. and Sagahyroon, A., 2021. A remote healthcare monitoring framework for diabetes prediction using machine learning. Healthcare Technology Letters, 8(3), pp.45-57.

[11] Chang, V., Bailey, J., Xu, Q.A. and Sun, Z., 2022. Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. Neural Computing and Applications, pp.1-17.

[12] Olisah, C.C., Smith, L. and Smith, M., 2022. Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective. Computer Methods and Programs in Biomedicine, 220, p.106773.

[13] Kibria, H.B., Nahiduzzaman, M., Goni, M.O.F., Ahsan, M. and Haider, J., 2022. An ensemble approach for the prediction of diabetes mellitus using a soft voting classifier with an explainable AI. Sensors, 22(19), p.7268.

[14] Tasin, I., Nabil, T.U., Islam, S. and Khan, R., 2023. Diabetes prediction using machine learning and explainable AI techniques. Healthcare Technology Letters, 10(1-2), pp.1-10.

[15] Annamalai, R. and Nedunchelian, R., 2021. Diabetes mellitus prediction and severity level estimation using OWDANN algorithm. Computational Intelligence and Neuroscience, 2021.

[16] Chang, V., Bailey, J., Xu, Q.A. and Sun, Z., 2022. Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. Neural Computing and Applications, pp.1-17.

[17] SVKR Rajeswari, V., 2021. Prediction of diabetes mellitus using machine learning algorithm. Annals of the Romanian Society for Cell Biology, pp.5655-5662.

[18] Parente, A. and Sutherland, J.C., 2013. Principal component analysis of turbulent combustion data: Data pre- processing and manifold sensitivity. Combustion and flame, 160(2), pp.340-350.

[19] Hasan, M.K., Alam, M.A., Das, D., Hossain, E. and Hasan, M., 2020. Diabetes prediction using ensembling of different machine learning classifiers. IEEE Access, 8, pp.76516-76531.

[20] Naz, H. and Ahuja, S., 2020. Deep learning approach for diabetes prediction using PIMA Indian dataset. Journal of Diabetes & Metabolic Disorders, 19, pp.391-403.

[21] Sannasi Chakravarthy, S.R., Bharanidharan, N., Vinothini, C. et al. Adaptive Mish activation and ranger optimizer-based SEA-ResNet50 model with explainable AI for multiclass classification of COVID-19 chest X-ray images. BMC Med Imaging 24, 206 (2024). https://doi.org/10.1186/s12880-024-01394-2

[22] Musthafa, M.M., T R, M., V, V.K. et al. Enhanced skin cancer diagnosis using optimized CNN architecture and checkpoints for automated dermatological lesion classification. BMC Med Imaging 24, 201 (2024). https://doi.org/10.1186/s12880-024-01356-8

[23] Mahesh T R, Muskan Gupta, Anupama T A, Vinoth Kumar V, Oana Geman, Dhilip Kumar V, An XAI-Enhanced EfficientNetB0 Framework for Precision Brain Tumor Detection in MRI Imaging,Journal of Neuroscience Methods, 2024,110227,ISSN 0165-0270,https://doi.org/10.1016/j.jneumeth.2024.110227.

[24] Kumaran S, Y., Jeya, J.J., R, M.T. et al. Explainable lung cancer classification with ensemble transfer learning of VGG16, Resnet50 and InceptionV3 using grad-cam. BMC Med Imaging 24, 176 (2024). https://doi.org/10.1186/s12880-024-01345-x

[25] Kudithi, T., Balajee, J., Sivakami, R. et al. Hybridized deep learning goniometry for improved precision in Ehlers-Danlos Syndrome (EDS) evaluation. BMC Med Inform Decis Mak 24, 196 (2024). https://doi.org/10.1186/s12911-024-02601-4

[26] Mahesh TR, Surbhi Bhatia Khan, A. Balajee, Ahlam Almusharraf, Thippa Reddy Gadekallu, Eid Albalawi, Vinoth Kumar; Water quality level estimation using IoT sensors and probabilistic machine learning model. Hydrology Research 2024; nh2024048. doi: https://doi.org/10.2166/nh.2024.048

[27] Natarajan, K., Vinoth Kumar, V., Mahesh, T.R. et al. Efficient Heart Disease Classification Through Stacked Ensemble with Optimized Firefly Feature Selection. Int J Comput Intell Syst 17, 174 (2024). https://doi.org/10.1007/s44196-024-00538-0

Edited by: Dhilip Kumar V
Special issue on: Unleashing the power of Edge AI for Scalable Image and Video Processing
Received: Sep 2, 2024
Accepted: Nov 8, 2024